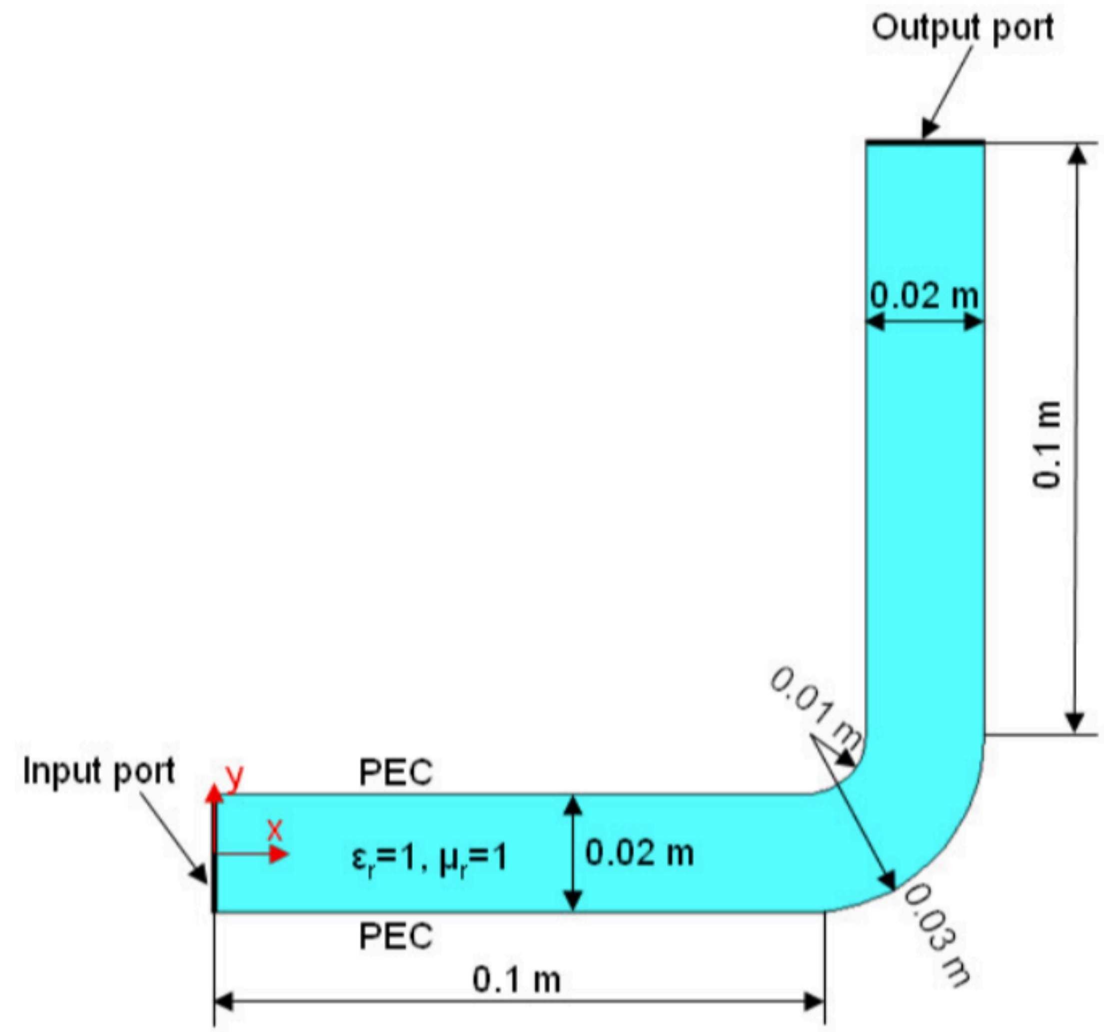
Project: 2-D DG-FEM Time domain Analysis of a Waveguide Bend

Starting from the available 2-D CG FEM frequency domain solver for waveguide problems (Moodle, Waveguide 90° Bend) it is necessary to develop a 2-D DG-FEM time domain solver with a given mesh and geometry. The expected outcomes of the project:

- Theoretical description of the 2-D DG-FEM in time domain for waveguide 90° bend analysis.
- Field computation for the first (dominant) even mode at the frequency within the single mode propagation range.
- Computation of the signal reflection (S11) and transmission (S21) parameters.
- Ez scalar field plot.
- H vector field plot.
- The corresponding Matlab code.



Boundary conditions:

$$\frac{\partial E_z}{\partial n}(x,y) + jk_x E_z(x,y) = 2jk_x C_1(y)\cdot e^{-jk_x x}, \text{ over the input port}$$

$$\frac{\partial E_z}{\partial n}(x,y) + jk_x E_z(x,y) = 0, \text{ over the output port}$$

$$E_z(x,y) = 0, \text{ PEC (waveguide wall)}$$

The absorbing boundary conditions are:

From the first-order Maxwell's curl equation-based absorbing boundary condition for the input port:

$$\hat{n} \times \vec{H} = \sqrt{\frac{\epsilon}{\mu}}\, \hat{n} \times (\hat{n} \times \vec{E}) - 2\sqrt{\frac{\epsilon}{\mu}}\, \hat{n} \times \left(\hat{n} \times \vec{E}_{\mathrm{inc}}\right)$$

and $\nabla \times \vec{E} = -\mu \frac{\partial}{\partial t} \vec{H}$

We get

$$\begin{aligned}
\hat{n} \times (\nabla \times \vec{E}) &= -\mu \frac{\partial}{\partial t} (\hat{n} \times \vec{H}) \\
&= -\mu \frac{\partial}{\partial t} \left[\sqrt{\frac{\epsilon}{\mu}}\, \hat{n} \times (\hat{n} \times \vec{E}) - 2\sqrt{\frac{\epsilon}{\mu}}\, \hat{n} \times \left(\hat{n} \times \vec{E}_{inc}\right)\right] \\
&= \sqrt{\epsilon\mu}\, \frac{\partial}{\partial t} \left[\hat{n} \times \vec{E} \times \hat{n} - 2\hat{n} \times \vec{E}_{inc} \times \hat{n}\right].
\end{aligned}$$

At input port,

$$n = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad \nabla \times \vec{E} = \begin{bmatrix} \frac{\partial E_z}{\partial y} \\ -\frac{\partial E_z}{\partial x} \\ 0 \end{bmatrix}, \quad \vec{n} \times \vec{E} \times \vec{n} = \begin{bmatrix} 0 \\ 0 \\ E_z \end{bmatrix}$$

Plug in, we get

$$\frac{\partial E_z}{\partial x} = \sqrt{\epsilon\mu} \frac{\partial}{\partial t} (2E_{\mathrm{inc}} - E_z) = \sqrt{\epsilon\mu} \frac{\partial}{\partial t} \left(2C_1(y) e^{-j\omega t} - E_z\right)$$

At output port, we have

$$\frac{\partial E_z}{\partial y} = -\sqrt{\epsilon\mu} \frac{\partial}{\partial t} E_z$$

Starting from Maxwell's equations

$$\nabla \times \vec{E} = -\mu \frac{\partial}{\partial t} \vec{H}$$

$$\nabla \times \vec{H} = \epsilon \frac{\partial}{\partial t} \vec{E}$$

$$\nabla \times (1): \ \nabla \times (\nabla \times \vec{E}) = -\mu \frac{\partial}{\partial t} (\nabla \times \vec{H}) = -\mu\epsilon \frac{\partial}{\partial t} \left(\frac{\partial}{\partial t} \vec{E} \right)$$

$$\nabla^2 \vec{E} - \mu\epsilon \frac{\partial^2}{\partial t} \vec{E} = 0$$

with polarization, E=Ez, one gets the differential equation:

$$\nabla(\alpha \nabla \phi) - \frac{\partial^2}{\partial t^2} \phi = 0, \text{ where } \alpha = \frac{1}{\mu\epsilon}, \ \phi = E_z$$

For DG-FEM, apply weighted residual method locally for each element:

$$\underbrace{\iint_{\Omega^e} w_i \nabla(\alpha \nabla \phi) d\Omega}_{\text{Integration by parts}} - \iint_{\Omega^e} w_i \frac{\partial^2}{\partial t^2} \phi \, d\Omega = 0$$

Integrate the first term by parts and use Gauss' Theorem:

$$\iint_{\Omega^e} \nabla w_i \alpha \nabla \phi \, d\Omega + \iint_{\Omega^e} w_i \frac{\partial^2}{\partial t^2} \phi \, d\Omega - \oint_{\partial \Omega^e} w_i \alpha \nabla \phi \cdot \vec{n} \, dl = 0$$

The line integral term is to be evaluated based on the type of edges of the element.

Approximation of unknown function: $\phi^e = \sum_{j=1}^{3} N_j^e \cdot \phi_j^e$

Plug in and replace weighting function by shape function:

$$\underbrace{\left[\sum_e \iint_{\Omega^e} \alpha \nabla N_i \nabla N_j \, d\Omega \right]}_{[M]} E_z + \underbrace{\left[\sum_e \iint_{\Omega^e} N_i N_j d\Omega \right]}_{[N]} \frac{d^2}{dt^2} E_z = \sum_{N_{edge} \in e} \int_{edge} w_i \alpha \nabla E_z \cdot \vec{n} \, dl$$

The RHS (line integral) term is to be evaluated based on the type of edges of the element, considering boundary conditions and numerical flux between the elements:

If the edge is inside the compuational domain (e.g. not on boundary), use central flux scheme:

$$\begin{aligned}\int_{edge} w_i \alpha \nabla E_z \cdot \vec{n}\, dl &= \int_{edge} w_i \alpha \frac{1}{2}\left(\nabla E_z^{int} + \nabla E_z^{ext}\right) \cdot \vec{n}\, dl \\ &= \int_{edge} \frac{1}{2} \alpha \nabla N_j \left(E_z^{int} + E_z^{ext}\right) \cdot \vec{n}\, N_i\, dl\end{aligned}$$

If the edge is on the input port, $\dfrac{\partial E_z}{\partial x} = \sqrt{\epsilon\mu}\dfrac{\partial}{\partial t}(2E_{\text{inc}} - E_z) = \sqrt{\epsilon\mu}\dfrac{\partial}{\partial t}(2C_1(y)\cos(\omega t) - E_z)$

$$\int_{edge} w_i \alpha \nabla E_z \cdot \vec{n} dl = \underbrace{\int_{edge} \frac{2}{\sqrt{\mu\epsilon}} \frac{\partial}{\partial t} C_1(y)\cos(\omega t) N_i dl}_{\{b\}} - \frac{\partial}{\partial t} E_z \underbrace{\left[\int_{edge} \frac{1}{\sqrt{\mu\epsilon}} N_i N_j dl\right]}_{[K]}$$

If the edge is on the ouput port,

$$\int_{edge} w_i \alpha \nabla E_z \cdot \vec{n} dl = -\frac{\partial}{\partial t} E_z \underbrace{\left[\int_{edge} \frac{1}{\sqrt{\mu\epsilon}} N_i N_j dl\right]}_{[K]}$$

If the edge is PEC, this term = 0.

Summing up, the integral contribution from edges is

$$\sum_{N_{edge} \in e} \int_{edge} w_i \alpha \nabla E_z \cdot \vec{n}\, dl = \sum_{edges} \{f\} + \{b\} - \frac{d}{dt} E_z [K]$$

Time stepping with backward difference:

$$\frac{d^2}{dt^2}E_z^t = \frac{E_z^t - 2E_z^{t-1} + E_z^{t-2}}{\Delta t^2}, \quad \frac{d}{dt}E_z^t = \frac{E_z^t - E_z^{t-1}}{\Delta t}$$

Plug in $$\underbrace{\left[\sum_e \iint_{\Omega^e} \alpha \nabla N_i \nabla N_j \, d\Omega\right]}_{[M]} E_z + \underbrace{\left[\sum_e \iint_{\Omega^e} N_i N_j d\Omega\right]}_{[N]} \frac{d^2}{dt^2} E_z = \sum_{N_{edge} \in e} \int_{edge} w_i \alpha \nabla E_z \cdot \vec{n} \, dl$$

$$[N]\left(\frac{E_z^t - 2E_z^{t-1} + E_z^{t-2}}{\Delta t^2}\right) + [M]E_z^t + [K]\left(\frac{E_z^t - E_z^{t-1}}{\Delta t}\right) = \sum_{edges} \{f\} + \{b\}$$

$$\left(\frac{1}{\Delta t^2}[N] + [M] + \frac{1}{\Delta t}[K]\right)E_z^t = \left(\frac{2}{\Delta t^2}[N] + \frac{1}{\Delta t}[K]\right)E_z^{t-1} - \frac{1}{\Delta t^2}[N]E_z^{t-2} + \sum_{edges}\{f\} + \{b\}$$

where $$[M] = \sum_e \iint_{\Omega^e} \frac{1}{\mu_0 \epsilon_0} \nabla N_i \nabla N_j \, d\Omega$$

$$[N] = \sum_e \iint_{\Omega^e} N_i N_j \, d\Omega$$

$$[K] = \int_{edge} \frac{1}{\sqrt{\mu\epsilon}} N_i N_j \, dl$$ if the element have an edge on the input/output port

$$\{b\} = \int_{edge} \frac{2}{\sqrt{\mu\epsilon}} \frac{d}{dt} C_1(y)^{\cos(wt)} N_i \, dl$$ if the element have an edge on the input port

$$\{f\} = \int_{edge} \frac{1}{2} \alpha \nabla N_j \left(E_z^{int} + E_z^{ext}\right) \cdot \vec{n} \, N_i \, dl$$ for edges that have a neighbour element

The outline of program:

```
load mesh;
assemble matrix [M], [N];
define constants and initialize;
time loop:
    element loop:
        check edge condition;
        if edge inside domain:
            add its contribution (flux term) to RHS;
        if edge on input port:
            add its contribution to [K] and [b];
        if edge on output port:
            add its contribution to [K];
        if edge on PEC:
            no contribution;
        solve Ez = left/right;
        slope limiter;
        calc Hx, Hy field;
    plot Ez, Hx, Hy fields
    calc S parameter
```